BAB 2
DASAR TEORI

2.1. KAJIAN PUSTAKA

Penelitian yang dilakukan oleh Purwanto pada tahun 2018 dengan judul "Analisis Kinerja *Dynamic Routing* Pada Protokol *Routing* EIGRP Untuk menentukan Jalur Terbaik Dengan DUAL" [2]**.** Dalam penelitiannya menggunakan metode rekayasa jaringan melalui kumpulan konsep, pendekatan dan metodologi, serta alat-alat bantu untuk merancang dan menginstalasi suatu kompleks sistem. Nilai *metric* serta parameter *bandwidth* dan *delay* digunakan sebagai acuan menentukkan jalur terbaik ke alamat *network* tujuan. Rekayasa jaringan menggunakan *Packet Tracer* untuk membuat simulasi jaringan dan *Axcenet Tools* untuk *tracing route*. Dari kesimpulan penelitian tersebut hasil parameter jalur kedua dari skenario dua lebih baik kemampuannya dalam mengirimkan data. Persamaan yang diteliti yaitu penggunan protokol *routing* EIGRP, perbedaannya pada skenario penulis yaitu melihat performansi ketika menerapkan *feasible successor* dengan yang tidak menerapkan *feasible successor*. Sedangkan peneliti hanya melihat perbandingan dari perhitungan nilai *metric*. Serta perbedaan pada *software* yang digunakan untuk melakukan simulasi jaringan.

Penelitian Lukman, dkk pada tahun 2019 dengan penilitian berjudul "Analisis Waktu Konvergensi *Routing* Protokol EIGRP Dan OSPF" membahas tentang perbandingan waktu konvergensi antara protokol *routing* EIGRP dan OSPF [4]. Perangkat lunak GNS3 digunakan untuk membuat perancangan dan simulasi *routing* EIGRP dan OSPF. Hasil penelitian dengan membandingkan waktu konvergensi pada dua metode protokol *routing* yaitu EIGRP dan OSPF menghasilkan bahwa waktu konvergensi untuk metode protokol *routing* EIGRP lebih baik atau lebih cepat dibanding metode protokol *routing* OSPF. Persamaan yang diteliti pada penggunaan protokol *routing* pada protokol *routing* EIGRP. Perbedaan pada skenario yang penulis lakukan adalah melihat performa protokol *routing* EIGRP dengan menerapkan *feasible successor* berdasarkan waktu konvergensi serta parameter QOS *delay, jitter, throughput,* dan *packet loss* sedangkan penelitian sebelumnya hanya melihat perbandingan waktu konvergensi.

Penelitian Sukmana pada tahun 2018 dengan penelitian berjudul "Simulasi Perbandingan *Multi-Protocol Label Switching* Terhadap *Routing Protocol* OSPF, RIPv2, Dan EIGRP" membahas tentang performa protokol *routing* menggunakan *service* MPLS dengan parameter QOS (*throughput*, *jitter*, *packet loss*, dan *ping response time*). Hasil penelitian menyimpulkan bahwa pengujian protokol *routing* dengan MPLS pada seluruh skenario memiliki nilai parameter QOS yang lebih baik dibandingkan dengan protokol *routing* tanpa MPLS. Persamaan yang diteliti pada penggunaan protokol *routing* EIGRP. Perbedaan pada skenario yang penulis uji yaitu melihat performa *feasible successor* yang terdapat pada protokol *routing* EIGRP berdasarkan parameter QOS dan waktu konvergensinya.

Tabel 2. 1 Parameter Penelitian

Penelitian Oleh	Parameter Penelitian				
	Tahun	Feasible Successor	EIGRP	Waktu Konvergensi	Parameter QOS
Timur Dali Purwanto	2018	✓	✓		
Lukman, Saputra, Pambudi	2019		✓	✓	
Dodi Sukmana	2018		✓	✓	✓
Penelitian Kami	2022	✓	✓	✓	✓

2.2. DASAR TEORI

2.2.1. Jaringan Komputer

Suatu kumpulan dari beberapa komputer (dan perangkat lainnya seperti *hub*, *switch*, dan *router*) yang saling terhubung satu sama lain melalui media perantara. Media perantara ini bisa berupa media kabel ataupun media tanpa kabel (nirkabel). Informasi dalam bentuk suatu data yang mengalir dari satu komputer ke komputer lainnya ataupun komputer ke perangkat lain, sehingga masing-masing dari komputer yang terhubung dapat saling bertukar data [5].

a) Router

Router merupakan sebuah perangkat jaringan yang bekerja pada lapisan *network* (model OSI) berfungsi untuk mengirim paket data dari suatu *network* sumber ke *network* tujuan. Router sering digunakan untuk menghubungkan

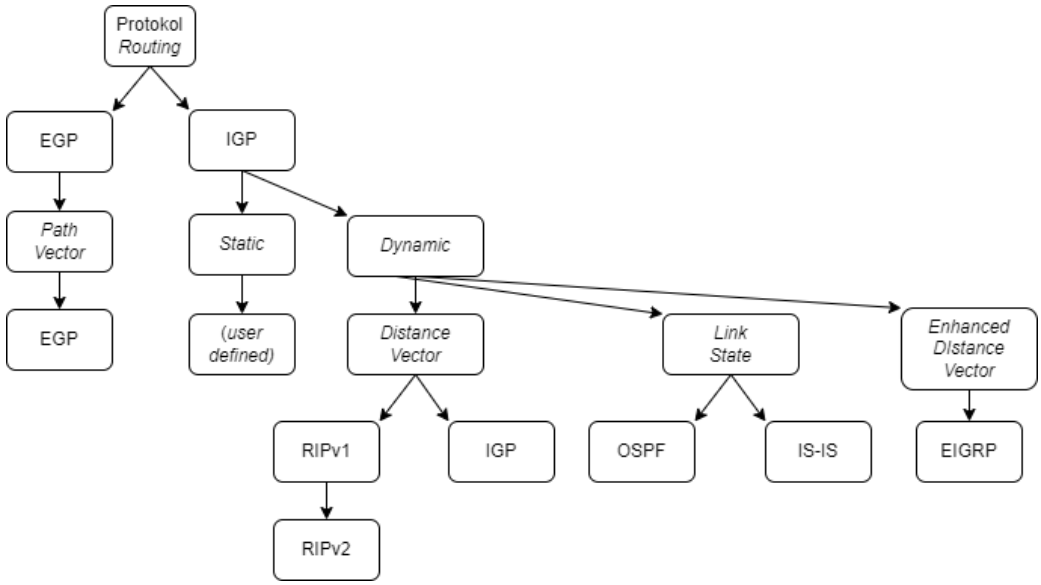
beberapa *network* baik yang sama ataupun berbeda dari segi topologinya. Router juga digunakan untuk membagi *network* besar menjadi beberapa sebuah *subnetwork* (*network-network* kecil) [6].

b) *Routing*

Sebuah router memiliki suatu kemampuan untuk mengetahui kemana rute perjalanan informasi akan dilewatkan dalam bentuk paket dengan nama *routing*. Istilah *routing* digunakan sebagai pemilihan jalur sebuah paket dari sebuah jaringan ke jaringan lainnya dimana jaringan tersebut saling terhubung melalui router yang menyimpan informasi tentang *routing* didalam *table routing.* Apabila paket-paket ditujukan untuk *host* lain yang berbeda *network* maka router akan meneruskannya ke *network* tujuan. Sebaliknya, jika paket-paket yang dituju untuk *host* yang berada dalam satu *network* maka router akan menghalangi paket keluar, sehingga paket-paket tersebut tidak membanjiri *network* yang lain [7,8].

2.2.2. Protokol *Routing*

Protokol *routing* pada jaringan komputer menjadi salah satu hal paling penting dalam menentukkan kualitas kinerja sebuah jaringan. Semakin cepat suatu protokol *routing* dalam menyampaikan sebuah informasi melalui *route* yang ada pada tabel *routing*, maka akan semakin baik kualitas dari jaringan terseut.



Gambar 2. 1 Protokol *Routing*

Protokol *routing* akan mendistribusikan tabel *routing*-nya sendiri ke router yang lain. Untuk menyesuaikan tabel *routing*, maka beberapa *routing* diberikan izin ke router-router yag lain untuk melakukan *update* secara berkala tentang status jaringan mereka. Secara umum ada dua kategori tabel protokol *routing* yaitu *routing* statis dan *routing* dinamis.

a) *Routing Static*

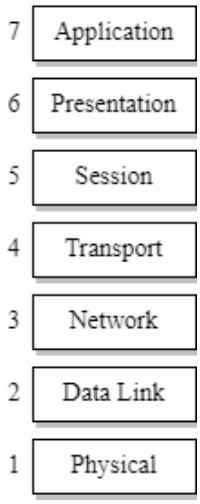
Routing static merupakan protokol *routing* yang masih memerlukan campur tangan administrator jaringan dan *routing* tersebut masih bekerja dengan cara manual untuk melakukan proses pertukaran informasi dengan router tetangganya. Sehingga membutuhkan waktu yang cukup lama untuk mengelola jaringan yang menggunakan *routing static* terutama kalau jaringannya mempunyai skala yang besar. *Routing static* tidak bisa memilih jalur yang optimal. Jaringan yang kemungkinan kecil mengalami perubahan dalam topologi biasanya menggunakan *routing* static.

b) *Routing Dynamic*

Routing dynamic merupakan protokol *routing* yang dapat menentukkan sendiri rute berdasarkan situasi dan kondisi setiap saat. *Routing dynamic* yang dibentuk oleh beberapa buah router yang masing-masing akan saling bertukar informasi pada router tetangganya menggunakan protokol *routing* untuk mengelola *routing table* secara dinamis, sehingga *routing dynamic* bisa tumbuh lebih cepat dan lebih luas. Protokol *routing* sendiri memiliki arsitektur yang berbeda-beda, sehingga kemampuan antar protokol *routing* akan berbeda juga [5,9].

2.3. MODEL *OPEN SYSTEM INTERCONNECTION* (OSI)

Secara umum model OSI membagi fungsi dari jaringan komputer menjadi tujuh lapisan. Model OSI telah menjadi referensi atau acuan umum yang digunakan sebagai pengembangan oleh para *developer* dan pembelajaran bagi siswa yang ingin memahami cara kerja jaringan komputer.



Gambar 2. 2 Model OSI

Model OSI memiliki tujuh lapisan yang berbeda dimana data dibagi menjadi unit data yang sesuai untuk setiap lapisan. Setiap unit disebut dengan *Protocol Data Unit* (PDU), diteruskan dari satu lapisan ke lapisan lain dalam perjalanannya ke atas atau ke bawah untuk melakukan komunikasi antar lapisan. Informasi pada beberapa lapisan akan mengalami perubahan bentuk yaitu terjadinya penambahan format atau pengalamatannya sendiri ke PDU yang disebut *header*. Proses terjadinya penambahan *header* ini disebut enkapsulasi. Sedangkan terjadinya proses pelepasan *header* disebut de-enkapsulasi [7,10].

Berikut adalah tujuh lapisan model OSI beserta fungsinya:

Tabel 2. 2 OSI *Layer* Dan Fungsinya

No	Nama Lapisan	PDU	Protocol	Perangkat	Fungsi
7	*Application*	Data	HTTP, FTP, SMTP, DHCP	Komputer	Menyediakan *user interface.*
6	*Presentation*	Data	*Redirectors*	-	Menyajikan data. Menangani enkripsi dan dekripsi.
5	*Session*	Data	DNS, *Authentication Protocol*	-	Memisahkan data dari berbagai aplikasi yang berbeda.

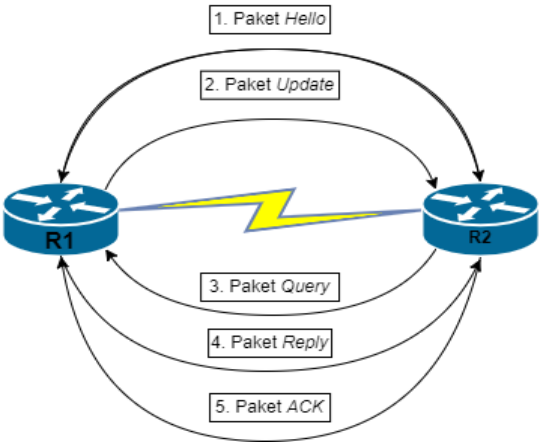
No	Nama Lapisan	PDU	Protocol	Perangkat	Fungsi
4	*Transport*	*Segment*	TCP, UDP	-	Menyediakan metode pengiriman data. Melakukan perbaikan kesalahan sebelum pengiriman ulang.
3	*Network*	*Packet*	IP ICMP ARP	Router, Firewall, Layer 3 Switch	Menangani *packet routing*, *logical addressing*.
2	*Data Link*	*Frame*	*Ethernet, token ring*, FDDI, NIC *driver*	Switch, NIC	Menyediakan akses ke media menggunakan MAC *address.* Melakukan error detection.
1	*Physical*	*Bits*	-	*Hub, Reapeater, Connector*	Mengatur koneksi *hardware.* Mengatur pengiriman dan penerimaan sinyal *binary.* Menangani *encoding bit.*

2.4. *ENHANCED INTERIOR GATEWAY ROUTING PROTOCOL* (EIGRP)

EIGRP merupakan sebuah protokol *routing* milik CISCO yang hanya bekerja pada router CISCO saja. Protokol *routing* EIGRP adalah protokol *distance vector* yang *classless*. Protokol *routing* EIGRP menawarkan fitur *backup router*. *Backup router* sendiri digunakan apabila ada perubahan informasi *network*, protokol *routing* EIGRP tidak harus melakukan kalkulasi ulang untuk menentukan *router* terbaik karena bisa langsung menggunakan *backup router.* Kalkulasi ulang *route* terbaik dilakukan jika *backup route* juga mengalami kegagalan [3].

2.4.1. PAKET-PAKET EIGRP

Dalam EIGRP terdapat beberapa jenis paket yang digunakan untuk membantu dalam berkomunikasi. Paket-paket tersebut dienkapsulasi oleh *Internet Protocol* (IP). Beberapa jenis paket tersebut, yaitu:



Gambar 2. 3 Paket-paket EIGRP

1) Paket Hello

Paket awal yang akan dikirimkan oleh router EIGRP adalah paket *hello* untuk mengecek keberadaan dirinya ke router tetangga. Sekumpulan paket *hello* dikirim secara berkala secara *multicast* dan tidak di-ACK secara langsung.

2) Paket Update

Paket *update* digunakan untuk memberikan informasi rute ke router tetangga yang tersimpan dalam tabel *routing*. Paket *update* akan dikirim secara *multicast* apabila terjadi perubahan konfigurasi atau topologi pada sebuah jaringan.

3) Paket *Query*

Paket yang berfungsi untuk menanyakan rute terbaik sebelumnya yang hilang. Jika paket update menemukan adanya rute yang hilang maka *multicast queries* akan dikirim untuk mencari informasi apakah router tetangga masih memiliki informasi rute tersebut. Apabila tidak mendapat respon, maka router yang kehilangan rute akan mencoba mengirimkan paket *unicast query*, satu-per satu ke setiap router tetangga hingga 16 kali pengulangan.

4) Paket *Reply*

Pengiriman paket *reply* terjadi digunakan sebagai respon untuk menjawab paket *query*. Setiap router yang menerima paket *query* akan merespon dengan mengirim paket *reply* secara unicast.

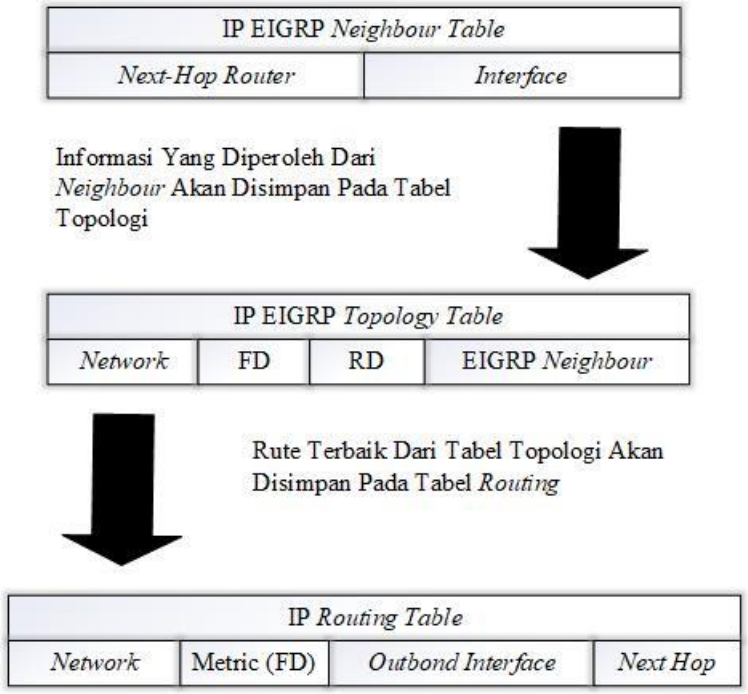
5) Paket *Acknowledgement* (ACK)

Paket yang digunakan untuk memberitahukan router pengirim bahwa

paket yang dikirimkan telah diterima oleh router tetangga. Setiap pengiriman paket *update, query,* dan *reply* perlu ada balasan paket ACK dari router lawan [1,7].

2.4.2. TABEL EIGRP

Sebuah router EIGRP akan menyimpan semua informasi tabel *routing* milik *neighbour*-nya pada tabel topologi. Sehingga router EIGRP dapat dengan cepat melakukan penyesuaian terhadap perubahan topologi jaringan. Hubungan antara ketiga tabel EIGRP dapat dilihat pada gambar 2.4.



Gambar 2. 4 Tabel EIGRP

Protokol *routing* EIGRP menyediakan beberapa buah tabel, yaitu:

1) Tabel Tetangga

Berisi informasi dari semua *adjacent* router atau router tetangga. Ketika sebuah router menemukkan router tetangga baru maka router akan menyimpan alamat router tetangga tersebut dan *interface* pada tabel tetangga.

2) Tabel Topologi

Tabel topologi berisi semua daftar jalur menuju ke semua *network* yang diketahui atau yang telah di *advertise* oleh router tetangga. Di dalamnya terdapat

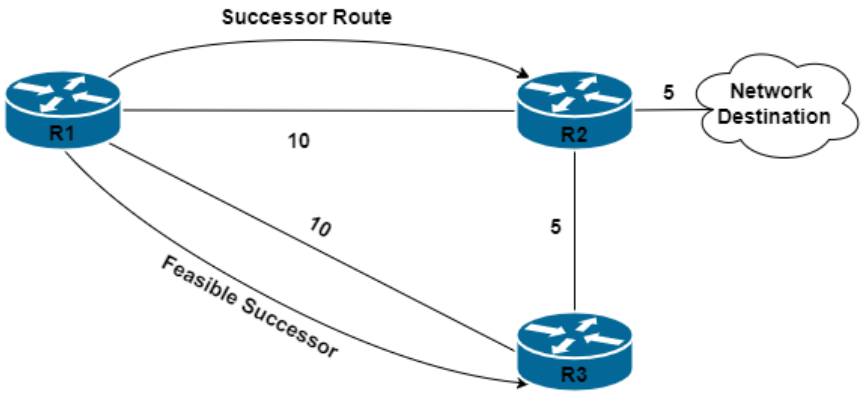
daftar semua *successor*, *feasible successor*, dan *feasible distance*. *Diffusing Update Algorithm* (DUAL) beraksi pada tabel topologi untuk menentukkan router sebagai *successor* dan *feasible successor* yang akan digunakan untuk pembentukan pada tabel *routing*. Sebuah entri pada tabel topologi dapat berisi dua kondisi, yaitu *active* (A) dan *passive* (P). Kondisi *passive* menunjukkan kondisi dimana router sedang tidak melakukan komputasi ulang. Sedangkan kondisi *active* menunjukan kondisi dimana router sedang melakukan komputasi atau kalkulasi ualng, yang artinya pada saat itu router melihat bahwa bahwa rute tujuan tidak ada *feasible successor*.

3) Tabel *Routing*

Berisi semua daftar *network* dan jalur terbaik. Tabel *Routing* juga bisa disebut sebagai tabel yang berisi daftar *successor*. Tabel *Routing* terbentuk setelah DUAL selesai dijalankan dan tabel topologi terbentuk [6].

2.5. KONSEP *DIFFUSING UPDATE ALGORITHM* (DUAL)

DUAL merupakan algoritma routing yang digunakan oleh *routing protocol* EIGRP. DUAL mendukung IP, IPX, dan *AppleTalk* yang merupakan protokol dari layer *network*. Dengan DUAL setiap router EIGRP dapat menentukkan jalur yang telah dikenal oleh router tetangga merupakan *looped* atau *free-loop*.



Gambar 2. 5 *Successor* Dan *Feasible Successor*

DUAL menggunakan beberapa istilah yang digunakan dalam menentukan jalur terbaik ke alamat *network* tujuan:

1) *Successor*

Successor adalah router tetangga yang digunakan sebagai jalan utama atau

jalur terbaik untuk meneruskan paket ke alamat *network* tujuan [1], [7].

2) *Feasible Successor* (FS)

Feasible Successor merupakan jalur *backup* atau jalur cadangan yang akan dipakai apabila router *successor* mengalami masalah atau *down*.

3) *Feasible distance* (FD)

Feasible distance merupakan nilai metric yang dihasilkan untuk menuju ke alamat *network* tujuan. Nilai metric tersebut akan menunjuk ke rute yang mempunyai nilai metric terkecil. Tentu saja untuk menuju ke alamat *network* tujuan tersebut akan melewati *successor*.

4) *Reported distance* (RD)

Reported distance merupakan nilai metric yang dilaporkan oleh router EIGRP tetangga ketika *successor* untuk menuju ke alamat *network* tujuan mengalami masalah.

5) *Feasible condition* (FC)

Feasible condition merupakan sebuah syarat yang harus dipenuhi apabila ada router tetangga yang ingin dijadikan sebagai *feasible successor*. Apabila nilai dari *Reported Distance* (RD) dari router lebih kecil dari nilai *Feasible Distance* (FD) dari *successor*, maka router tetangga tersebut layak untuk dijadikan sebagai *feasible successor* [1,7].

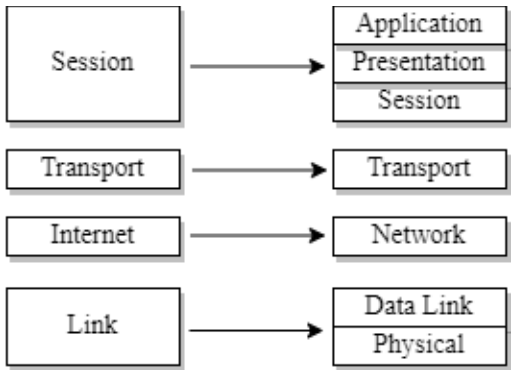
2.5.1. *FEASIBLE SUCCESSOR* (FS)

Feasible Successor adalah jalur backup dari *successor* (dengan memenuhi syarat nilai AD dari *feasible successor* harus lebih kecil daripada nilai FD dari *successor*). Ketika router menemukan *neighbour* baru, maka router akan mengirimkan sebuah *update* mengenai rute-rute yang ia ketahui kepada *neighbour* baru tersebut dan juga sebaliknya menerima informasi yang sama dari *neighbour*. *Update* ini lah yang akan membangun tabel topologi. Tabel topologi berisi informasi semua *network destination* yang di *advertise* oleh router *neighbour*. Jika jalur *successor* mengalami masalah algoritma *routing* DUAL tidak perlu melakukan proses perhitungan ulang untuk menemukan rute baru. DUAL cukup menggunakan *feasible successor* sebagai jalur cadangan untuk meneruskan paket ke alamat *network* yang dituju sehingga peran *successor* akan digantikan oleh *feasible successor* [1].

2.6. MODEL *TRANSMISSION CONTROL PROTOCOL / INTERNET PROTOCOL* (TCP/IP)

Pada tahun 1983 TCP/IP merupakan pengganti *Network Control Protocol* (NCP) dan diberi otorisasi sebagai sarana tranportasi data resmi untuk segala hal yang berhubungan dengan ARPANET.

Model TCP/IP merupakan versi ringkas dari model OSI. Model OSI memiliki 7 lapisan sedangkan model TCP/IP hanya memiliki 5 lapisan. Berikut adalah perbandingan antara model TCP/IP dan Model OSI:



Gambar 2. 6 Gambar Perbandingan Model TCP/IP dan OSI (kiri TCP/IP, kanan OSI)

2.6.1. *TRANSMISSION CONTROL PROTOCOL* (TCP)

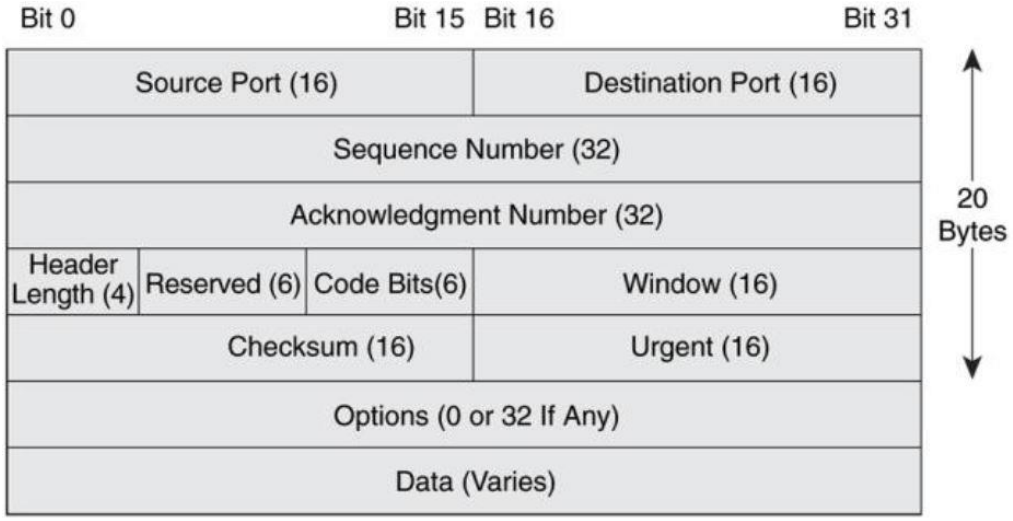
Transmission control protocol (TCP) merupakan salah satu dari protokol yang berada pada lapisan *transport* model OSI atau model TCP/IP. TCP memiliki beberapa karakteristik yaitu protokol yang bersifat *connection-oriented protocol* yang artinya sebelum proses transmisi data terjadi dua aplikasi TCP harus melakukan pertukaran kontrol informasi (*handshaking*), TCP juga merupakan protokol yang bersifat *reliable* karena menerapkan fitur deteksi kesalahan dan *retransmission* apabila ada data yang rusak sehingga kebutuhan data dapat terjamin, dan *byte stream service* artinya paket akan dikirimkan ke tujuan secara berurutan (*sequencing*) [6].

Protokol TCP bertanggung jawab dalam hal pengiriman data dari sumber ke tujuan. TCP dapat mendeteksi suatu kesalahan atau hilangnya data dan melakukan

pengiriman kembali sampai data diterima dengan lengkap. TCP terus meminta konfirmasi setiap kali data dikirim untuk memastikan apakah data telah sampai di tempat tujuan. Kemudian TCP mengirimkan data berikutnya atau pengiriman ulang jika data sebelumnya tidak sampai atau rusak. Data yang dikirim dan diterima kemudian diatur berdasarkan nomor urut [9].

2.6.2. *Header* TCP

TCP memiliki ukuran *header* yang bervariasi dimana terdiri atas beberapa *field*. Ukuran segmen pada TCP *header* yaitu paling kecil 20 *Bytes* pada IPv4 dan 40 *Bytes* untuk IPv6. Pada gambar 2.7 dibawah ini merupakan *field* dan ukuran pada TCP *header*.

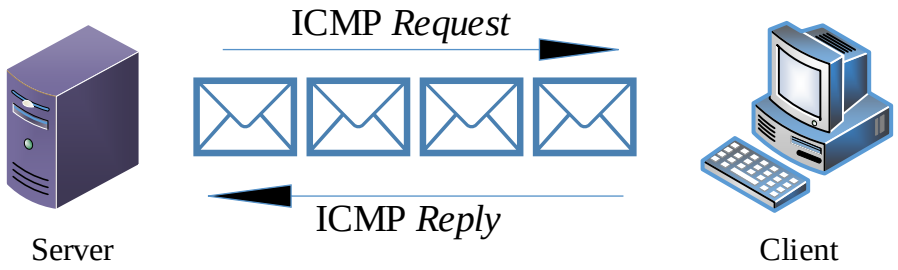


Gambar 2. 7 TCP *Segment*

Pada saat melakukan transmisi data akan dibagi menjadi bagian-bagian yang disebut segmen, dimana pada setiap tansmisi *Maximum Transmission Unit* (MTU) berukuran 1500 *Bytes* untuk akses jaringan komputer menggunakan *ethernet*. Sehingga jumlah data yang terdapat pada segmen TCP MTU tersebut dikurangi dengan ukuran TCP *header* dan ukuran IP *header*. *Maximum Segment Size* (MSS) merupakan total data yang terdapat pada segmen TCP. Dapat diumpamakan apabila digunakan *ethernet* IPv4, MSS atau jumlah data maksimal yang terdapat pada segmen TCP adalah sebesar 1460 *Bytes* [10].

2.7. *INTERNET CONTROL MESSAGE PROTOCOL* (ICMP)

Internet Control Message Protocol (ICMP) merupakan suatu protokol yang digunakan untuk memperoleh status dari suatu perangkat jaringan dengan mengirimkan pesan-pesan khusus yang dapat memicu pesan *reply* dari perangkat jaringan komputer.



Gambar 2. 8 Mekanisme Paket ICMP

Pada kondisi perangkat jaringan normal perangkat komputer dapat melakukan operasi dengan mmanfaatkan infrastruktur komunikasi. Namun ada beberapa kondisi dimana ketika koneksi jaringan terganggu, misal akibat komputer *crash*, putusnya jalur komunikasi, atau perangkat jaringan mati. Pada situasi tersebut, protokol ICMP membantu untuk mendapatkan status dari perangkat-perangkat jaringan dengan mengirimkan *request* kepada perangkat tujuan. Perangkat tujuan jika dalam kondisi baik maka akan merespon pesan tersebut, sehingga komputer monitoring dapat mengambil kesimpulan bahwa perangkat tersebut berjalan dengan normal [11].

2.8. VIRTUALISASI

Virtualisasi merupakan payung besar dari teknologi dan konsep yang dimaksudkan untuk menyediakan lingkungan abstrak baik perangkat keras *virtual* atau sistem operasi untuk menjalakan aplikasi. Virtualisasi adalah kombinasi rekayasa perangkat keras dan perangkat lunak yang menciptakan *Virtual Machine* (VM) dan memungkinkan beberapa sistem operasi berjalan pada perangkat fisik yang sama. Tujuan virtualisasi adalah untuk meningkatkan kinerja, ketersediaan, kecepatan, pengelolaan, hingga keamanan [12].

2.8.1. *ORACLE* VM *VIRTUALBOX*

Oracle VM VirtualBox merupakan sebuah perangkat lunak virtualisasi yang digunakan untuk melakukan operasi sistem "tambahan" pada sistem "utama". *Oracle* VM *VirtualBox* juga termasuk sebuah aplikasi virtualisasi lintas *platform* yang di instal pada sebuah komputer berbasis *Intel* maupun AMD. Hal ini memperluas kemampuan komputer hingga dapat mengoperasikan banyak sistem operasi pada beberapa *Virtual Machine* (VM) bersamaan [12].

Gambar 2. 9 Logo *Oracle VM VirtualBox*

2.9. *NETWORK EMULATOR*

Network emulator adalah perangkat lunak yang menjalankan perangkat virtual sama persis seperti perangkat asli di dunia nyata. Sistem operasi serta *images* dari perangkat jaringan yang dijalankan secara *virtual* oleh *network emulator* sama seperti dengan aslinya sehingga beban mesin yang menjalankan *emulator* menjadi berat. Meskipun *network emulator* memakai *resource* komputer yang besar, *emulator* dapat mengemulasikan *network* yang lebih nyata dibandingkan *network simulator* [9].

2.9.1. *GRAPHICAL NETWORK SIMULATOR* (GNS3)

Graphical Network Simulator (GNS3) adalah emulator *open source* yang digunakan untuk mensimulasikan jaringan dimana membutuhkan CISCO IOS (*Internetwork Operating System*) untuk memungkinkan kombinasi *virtual* dan perangkat nyata melakukan emulasi jaringan. CISCO IOS berjalan secara *virtual* di laptop ataupun komputer. GNS3 adalah *graphical front end* untuk *dynagen* yang berjalan di atas *dynamips* yang merupakan program inti untuk membuat emulasi

CISCO IOS menjadi mungkin dan menyediakan antarmuka berbasis teks yang ramah pengguna [9,10].

Gambar 2. 10 Logo Perangkat Lunak GNS3

2.10. *DISTRIBUTED INTERNET TRAFFIC* (D-ITG)

D-ITG merupakan perangkat lunak yang mampu menghasilkan lalu lintas IPv4 dan IPv6 dengan mereplikasi beban kerja aplikasi internet secara akurat. D-ITG sendiri merupakan alat pengukur jaringan yang dapat mengukur kinerja paling umum (misalnya *throughput, delay, jitter,* dan *packet loss*) pada tingkat paket. Pada lapisan *transport* perangkat lunak D-ITG saat ini sudah mendukung *Transmission Control Protocol* (TCP), *User Datagram Protocol* (UDP), *Strem Control Transmission Protocol* (SCTP1), dan *Datagram Congestion Control Protocol* (DCCP1) [15].

Tabel 2. 3 Perintah yang digunakan pada D-ITG

Perintah	Keterangan
-T	Mengatur *transmission protocol*
-t	Waktu yang dibutuhkan selama pengujian 60s
-C	Paket konstan yang dikirimkan pada tiap detik (*default*: 1000 paket/*second*)
-c	Ukuran paket yang akan dialirkan pada pengujian (*Bytes*)
-a	Alamat IP *address* yang dituju sebagai *server*
./ITGRecv	Menjadikan suatu perangkat sebagai penerima
./ITGSend	Perintah untuk mengirim data dari sisi pengirim

2.11. *WIRESHARK*

Perangkat lunak *Wireshark* merupakan *packet analyser* gratis dan *open source* yang digunakan untuk *troubleshooting* jaringan, analisis perangkat lunak, dan pendidikan. *Wireshark* memiliki fitur standar untuk melakukan sebuah protokol *analyser* dan beberapa fitur yang tidak ada pada produk yang lain [16]. Perangkat

lunak *wireshark* digunakan untuk merekam semua alur paket yang lewat saat menjalankan skenario pengujian serta menampilkan data tersebut sedetail mungkin untuk digunakan sebagai bahan analisis dalam penelitian.

2.12. KONVERGENSI

Konvergensi merupakan suatu bahasan dalam *dynamic routing* yang mempunyai keadaan dimana ketika semua router telah mempunyai *routing table* mereka sendiri secara tetap dan kosisten. Jaringan yang konvergensi ketika semua router telah mendapatkan hasil lengkap dan akurat mengenai informasi jaringan. Waktu konvergensi adalah waktu saat semua router berbagi informasi, menghitung jalur terbaik, dan memperbarui *routing table* mereka. Jaringan tidak akan berhenti beroperasi sampai semua *network* mendapatkan status konvergensi. Pencapaian status konvergensi secara tepat menandakan protokol *routing* yang lebih baik.

Rumus untuk mendapatkan waktu konvergensi yang diinginkan adalah sebagai berikut[17]:

Konvergensi= waktu ACK diterima saat *link up* – ACK terakhir saat *link down* (2.1)

2.13. *QUALITY OF SERVICE* (QOS)

Quaility of Service (QoS) adalah kemampuan yang dimiliki suatu jaringan untuk menyediakan layanan yang baik dengan menyediakan *bandwidth*, mengatasi *jitter* dan *delay* [18]. QoS merupakan sebuah sistem arsitektur *end to end* dan bukan merupakan sebuah fitur yang terdapat pada jaringan. QoS pada sebuah network merujuk pada tingkat kecepatan dan kehandalan penyampaian berbagai jenis beban data di dalam suatu komunikasi [19].

Parameter-parameter *Quality of Service* (QoS) merupakan kumpulan performansi dari besaran teknis, yaitu:

1) *Delay*

Waktu yang dibutuhkan data untuk menempuh jarak dari asal ke tujuan. *Delay* dapat dipengaruhi oleh jarak, media, fisik, kongesti atau juga waktu proses yang lama [20]. Persamaan perhitungan *delay*:

$$Delay = \frac{\text{Total } delay}{\text{Total Paket Yang Diterima}} \quad (2.2)$$

2) *Jitter*

Jitter bisa disebut juga variasi *delay*. Hal ini diakibatkan terjadinya variasi-variasi dalam panjang antrian, dalam waktu pengolahan data, dan juga dalam waktu penghimpunan ulang paket-paket di akhir perjalanan *jitter*[20]. Persamaan perhitungan jitter:

$$Jitter = \frac{\text{Total variasi } delay}{\text{Total Paket Yang Diterima}} \quad (2.3)$$

3) *Throughput*

Suatu kecepatan transfer data efektif yang diukur dalam *bit per sceond* (bps). *Throughput* merupakan jumlah total kedatangan paket yang sukses diamati pada tujuan selama interval waktu tertentu dibagi oleh durasi interval waktu tersebut[20]. Persamaan perhitungan *throughput*:

$$Throughput = \frac{Paket\ Data\ Diterima}{Lama\ Pengamatan} \quad (2.4)$$

4) *Packet Loss*

Packet Loss adalah suatu parameter yang menggambarkan suatu kondisi yang menunjukkan jumlah total paket yang hilang dapat terjadi karena *collision* dan *congestion* pada jaringan[21]. Persamaan perhitungan *Packet Loss*:

$$Packet\ Loss = \frac{Paket\ Data\ Yang\ Hilang}{Paket\ Data\ Yang\ Dikirimkan} \times 100\% \quad (2.5)$$